How Good is my Learning Algorithm? Building Cross-Validation Confidence Intervals for Test Error

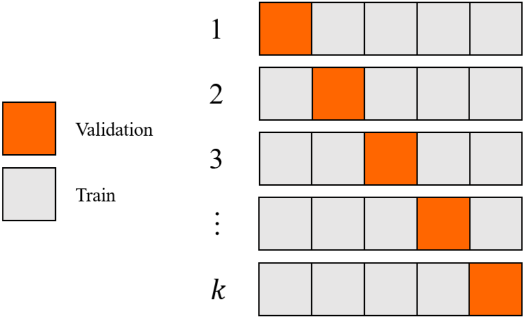
Alexandre Bayle

Department of Statistics, Harvard University

Joint work with Pierre Bayle, Lucas Janson, Lester Mackey

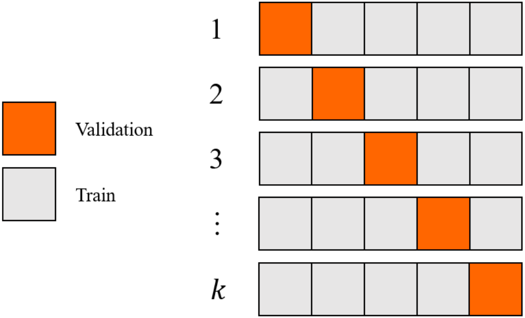
Stat 310 – April 3, 2024

Cross-validation (CV)



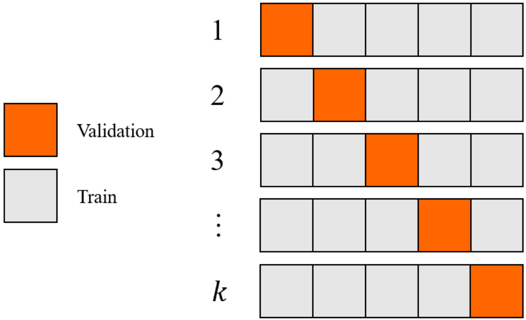
- Divide data into k folds

Cross-validation (CV)



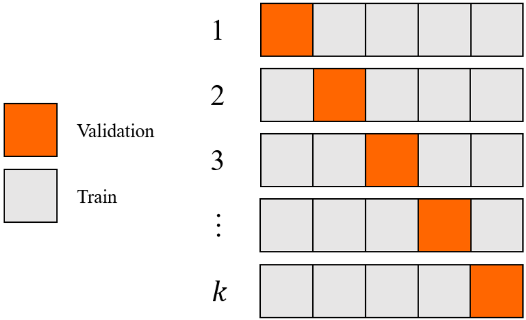
- Divide data into k folds
- Fit k prediction rules, each with one fold held out

Cross-validation (CV)



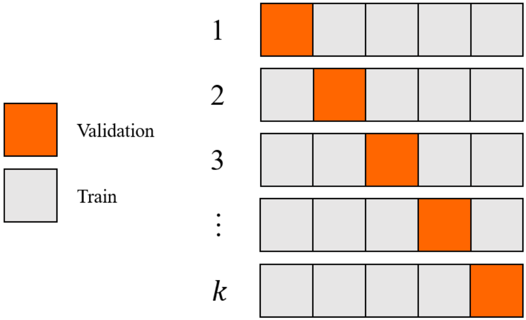
- Divide data into k folds
- Fit k prediction rules, each with one fold held out
- Evaluate each prediction rule on its held-out set

Cross-validation (CV)



- Divide data into k folds
- Fit k prediction rules, each with one fold held out
- Evaluate each prediction rule on its held-out set
- Average the k error estimates

Cross-validation (CV)



- Divide data into k folds
- Fit k prediction rules, each with one fold held out
- Evaluate each prediction rule on its held-out set
- Average the k error estimates

Pros:

- Unbiased for test error
- Lower variance than single train-test split

How good is my learning algorithm?

How good is my learning algorithm?

Need: Test error confidence intervals to quantify uncertainty

How good is my learning algorithm?

Need: Test error confidence intervals to quantify uncertainty

Problem: Existing intervals often invalid & CV distribution is complex

Is algorithm A actually better than algorithm B?

Is algorithm A actually better than algorithm B?

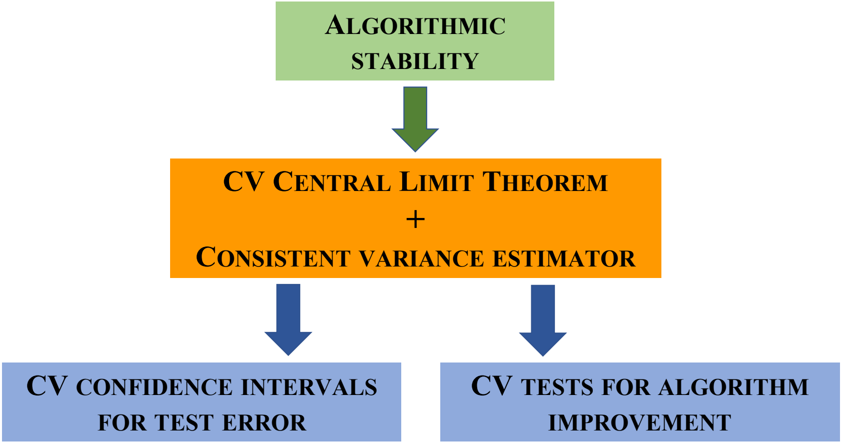
Need: Trustworthy hypothesis tests of algorithm improvement

Is algorithm A actually better than algorithm B?

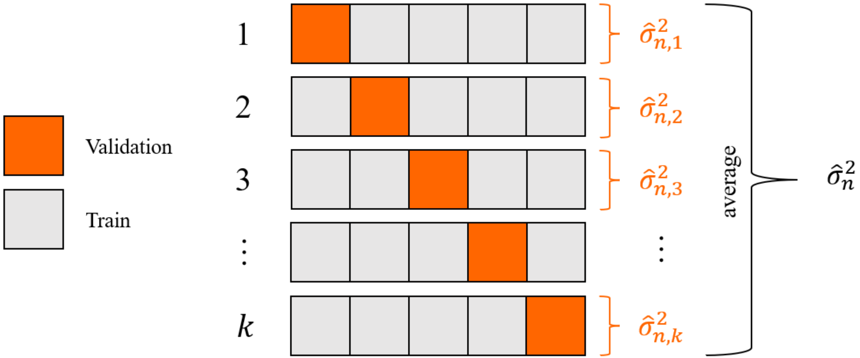
Need: Trustworthy hypothesis tests of algorithm improvement

Problem: Standard tests (like the cross-validated t-test, the repeated train-validation t-test, and the 5×2-fold CV test) do not appropriately account for dependence and have no correctness guarantees

Our Contributions



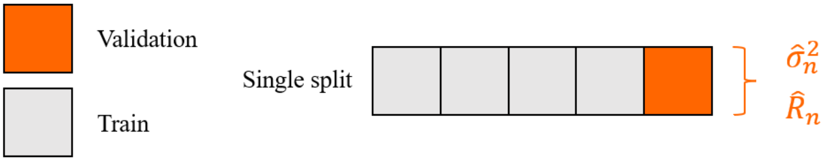
CV CLT



$$Z = \frac{\sqrt{n} \cdot \hat{R}_n}{\hat{\sigma}_n}$$

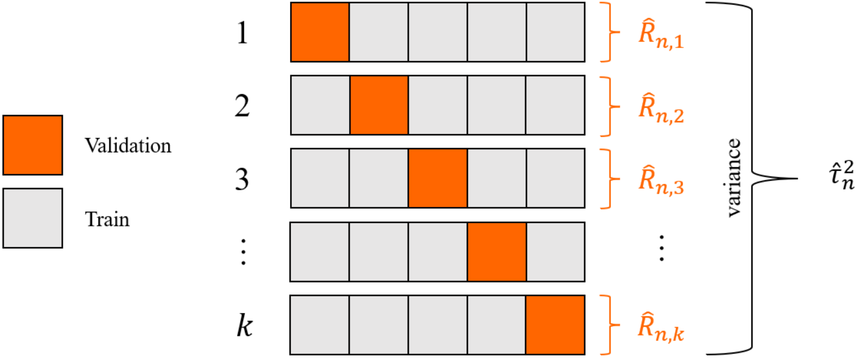
(Bayle, Bayle, Janson, and Mackey, 2020)

Hold-out CLT



$$Z = \frac{\sqrt{n/k} \cdot \hat{R}_n}{\hat{\sigma}_n}$$

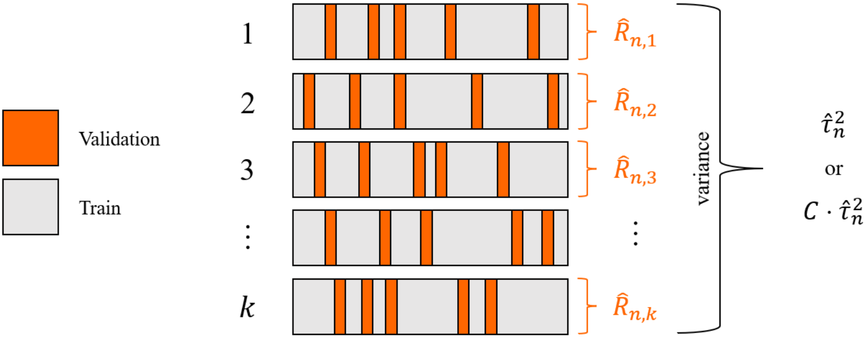
CV t



$$t = \frac{\frac{1}{k}\sum_{i=1}^{k} \hat{R}_{n,i}}{\hat{\tau}_n/\sqrt{k}}$$

(Dietterich, 1998)

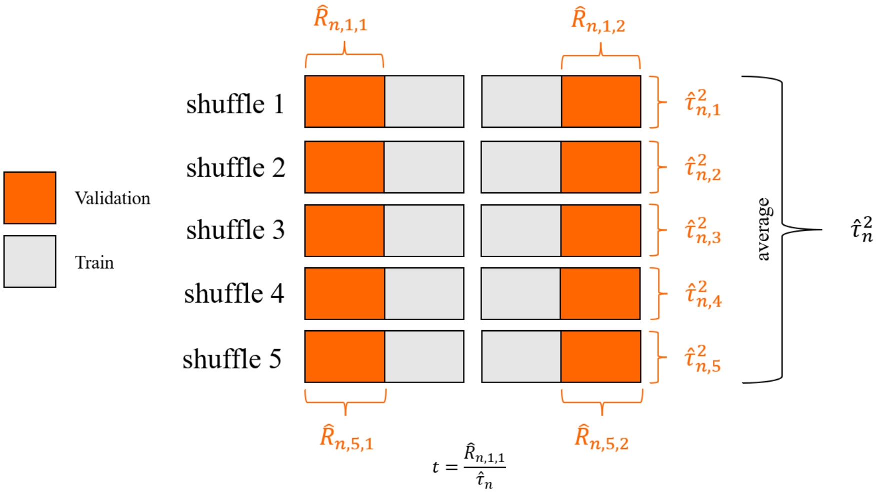
(Corrected) repeated train-validation t



$$t = \frac{\frac{1}{k}\sum_{i=1}^{k} \hat{R}_{n,i}}{\hat{\tau}_n/\sqrt{k}} \qquad C = 1 + k\frac{n_{test}}{n_{train}}$$

(Nadeau and Bengio, 2003)

5x2 CV



(Dietterich, 1998)

Notations and Problem Setup

- **Datapoints $Z_1, \ldots, Z_n$**
  - Often each $Z_i = (X_i, Y_i)$ with covariates X_i and response Y_i
  - For any vector B of indices, Z_B denotes the corresponding vector of datapoints

Notations and Problem Setup

- **Datapoints $\boldsymbol{Z_1, \ldots, Z_n}$**
  - Often each $Z_i = (X_i, Y_i)$ with covariates X_i and response Y_i
  - For any vector B of indices, Z_B denotes the corresponding vector of datapoints
- **Loss function $\boldsymbol{h_n(Z_i, Z_B)}$**: error when training on Z_B and testing on Z_i

Notations and Problem Setup

- **Datapoints $\boldsymbol{Z_1, \ldots, Z_n}$**
  - Often each $Z_i = (X_i, Y_i)$ with covariates X_i and response Y_i
  - For any vector B of indices, Z_B denotes the corresponding vector of datapoints
- **Loss function $\boldsymbol{h_n(Z_i, Z_B)}$**: error when training on Z_B and testing on Z_i
  - **Regression:** $h_n(Z_i, Z_B) = (Y_i - \widehat{f}(X_i; Z_B))^2$ for $\widehat{f}(\cdot; Z_B)$ trained on Z_B

Notations and Problem Setup

- **Datapoints $\boldsymbol{Z_1, \ldots, Z_n}$**
  - Often each $Z_i = (X_i, Y_i)$ with covariates X_i and response Y_i
  - For any vector B of indices, Z_B denotes the corresponding vector of datapoints
- **Loss function $\boldsymbol{h_n(Z_i, Z_B)}$**: error when training on Z_B and testing on Z_i
  - **Regression:** $h_n(Z_i, Z_B) = (Y_i - \widehat{f}(X_i; Z_B))^2$ for $\widehat{f}(\cdot; Z_B)$ trained on Z_B
  - **Classification:** $h_n(Z_i, Z_B) = \mathbb{1}[Y_i \neq \widehat{f}(X_i; Z_B)]$

Notations and Problem Setup

- **Datapoints $Z_1, \ldots, Z_n$**
  - Often each $Z_i = (X_i, Y_i)$ with covariates X_i and response Y_i
  - For any vector B of indices, Z_B denotes the corresponding vector of datapoints
- **Loss function $h_n(Z_i, Z_B)$**: error when training on Z_B and testing on Z_i
  - **Regression:** $h_n(Z_i, Z_B) = (Y_i - \widehat{f}(X_i; Z_B))^2$ for $\widehat{f}(\cdot; Z_B)$ trained on Z_B
  - **Classification:** $h_n(Z_i, Z_B) = \mathbb{1}[Y_i \neq \widehat{f}(X_i; Z_B)]$
- **Validation sets $\{B'_j\}_{j=1}^k$** and associated **training sets $\{B_j\}_{j=1}^k$**
  - Validation sets partition datapoint indices $\{1, \ldots, n\}$ into k folds
  - k can be fixed or grow with n

Notations and Problem Setup

- **Datapoints $\boldsymbol{Z_1, \ldots, Z_n}$**
  - Often each $Z_i = (X_i, Y_i)$ with covariates X_i and response Y_i
  - For any vector B of indices, Z_B denotes the corresponding vector of datapoints
- **Loss function $\boldsymbol{h_n(Z_i, Z_B)}$**: error when training on Z_B and testing on Z_i
  - **Regression:** $h_n(Z_i, Z_B) = (Y_i - \widehat{f}(X_i; Z_B))^2$ for $\widehat{f}(\cdot; Z_B)$ trained on Z_B
  - **Classification:** $h_n(Z_i, Z_B) = \mathbb{1}[Y_i \neq \widehat{f}(X_i; Z_B)]$
- **Validation sets $\boldsymbol{\{B'_j\}_{j=1}^k}$** and associated **training sets $\boldsymbol{\{B_j\}_{j=1}^k}$**
  - Validation sets partition datapoint indices $\{1, \ldots, n\}$ into k folds
  - k can be fixed or grow with n

Cross-validation (CV) error

$$\boldsymbol{\widehat{R}_n = \frac{1}{n} \sum_{j=1}^{k} \sum_{i \in B'_j} h_n(Z_i, Z_{B_j})}$$

Why CV Error?

Cross-validation error: $\widehat{R}_n = \frac{1}{n}\sum_{j=1}^{k}\sum_{i\in B'_j} h_n(Z_i, Z_{B_j})$

Why CV Error?

Cross-validation error: $\widehat{R}_n = \frac{1}{n}\sum_{j=1}^{k}\sum_{i\in B_j'} h_n(Z_i, Z_{B_j})$

k**-fold test error:** $R_n = \frac{1}{n}\sum_{j=1}^{k}\sum_{i\in B_j'} \mathbb{E}[h_n(Z_i, Z_{B_j}) \mid Z_{B_j}]$
$= \frac{1}{k}\sum_{j=1}^{k} \mathbb{E}[h_n(Z_0, Z_{B_j}) \mid Z_{B_j}]$

- Average test error of the k prediction rules $\widehat{f}(\cdot; Z_{B_j})$
- Common inferential target

Why CV Error?

Cross-validation error: $\widehat{R}_n = \frac{1}{n}\sum_{j=1}^{k}\sum_{i\in B'_j} h_n(Z_i, Z_{B_j})$

k-**fold test error:** $R_n = \frac{1}{n}\sum_{j=1}^{k}\sum_{i\in B'_j} \mathbb{E}[h_n(Z_i, Z_{B_j}) \mid Z_{B_j}]$
$= \frac{1}{k}\sum_{j=1}^{k} \mathbb{E}[h_n(Z_0, Z_{B_j}) \mid Z_{B_j}]$

- Average test error of the k prediction rules $\widehat{f}(\cdot; Z_{B_j})$
- Common inferential target

Goal: Establish a central limit theorem for $\widehat{R}_n - R_n$

Application: Confidence Intervals for Test Error

Problem

Construct an asymptotically-exact $(1-\alpha)$-confidence interval for k-fold test error R_n

Application: Confidence Intervals for Test Error

Problem

Construct an asymptotically-exact $(1-\alpha)$-confidence interval for k-fold test error R_n

Solution: CV Confidence Interval for Test Error

If we have a CLT and a variance estimator $\widehat{\sigma}_n^2$ that satisfies relative error consistency ($\widehat{\sigma}_n^2/\sigma_n^2 \xrightarrow{p} 1$), then the interval

$$C_\alpha = \widehat{R}_n \pm q_{1-\alpha/2}\,\widehat{\sigma}_n/\sqrt{n}$$

satisfies

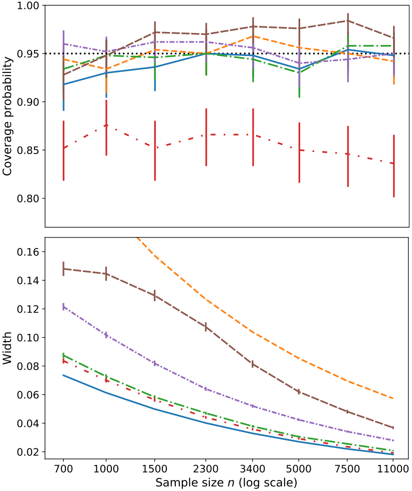
$$\lim_{n\to\infty} \mathbb{P}(R_n \in C_\alpha) = 1-\alpha$$

where $q_{1-\alpha/2}$ is the $(1-\alpha/2)$-quantile of a standard normal distribution

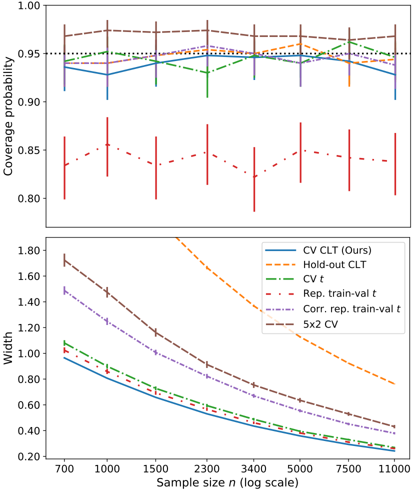
Confidence Intervals for Test Error, $k=10$

$$C_\alpha = \widehat{R}_n \pm q_{1-\alpha/2}\,\widehat{\sigma}_n/\sqrt{n} \quad \text{with} \quad 1-\alpha = 0.95$$

Our CV CLT procedure: valid coverage, smallest width



ℓ^2-regularized logistic regression (Higgs)



Random forest regression (FlightDelays)

Application: CIs for Test Error Difference

Problem

Construct an asymptotically-exact $(1 - \alpha)$-confidence interval for the difference in k-fold test errors

Application: CIs for Test Error Difference

Problem

Construct an asymptotically-exact $(1-\alpha)$-confidence interval for the difference in k-fold test errors

Solution: CV Confidence Interval for Test Error Difference

For a target loss function ℓ, define the $\mathcal{A}_1$-$\mathcal{A}_2$ loss difference

$$h_n(Z_0, Z_B) = \ell(Y_0, \widehat{f}_1(X_0; Z_B)) - \ell(Y_0, \widehat{f}_2(X_0; Z_B)),$$

if we have a CLT and a variance estimator $\widehat{\sigma}_n^2$ that satisfies relative error consistency ($\widehat{\sigma}_n^2/\sigma_n^2 \xrightarrow{p} 1$), then the interval

$$C_\alpha = \widehat{R}_n^{(1)} - \widehat{R}_n^{(2)} \pm q_{1-\alpha/2}\, \widehat{\sigma}_n/\sqrt{n}$$

satisfies

$$\lim_{n\to\infty} \mathbb{P}(R_n^{(1)} - R_n^{(2)} \in C_\alpha) = 1 - \alpha$$

Application: Tests for Algorithm Improvement

Problem

Construct an asymptotically-exact level α test of whether $\mathcal{A}_1$ has smaller k-fold test error than $\mathcal{A}_2$

Application: Tests for Algorithm Improvement

Problem

Construct an asymptotically-exact level α test of whether $\mathcal{A}_1$ has smaller k-fold test error than $\mathcal{A}_2$

Solution: CV Test for Improved Test Error

For a target loss function ℓ, define the $\mathcal{A}_1$-$\mathcal{A}_2$ loss difference

$$h_n(Z_0, Z_B) = \ell(Y_0, \widehat{f}_1(X_0; Z_B)) - \ell(Y_0, \widehat{f}_2(X_0; Z_B)),$$

and consider testing $H_0 : R_n \geq 0$ ($\mathcal{A}_1$ not better) against $H_1 : R_n < 0$ ($\mathcal{A}_1$ is better). If we have a CLT and a variance estimator $\widehat{\sigma}_n^2$ that satisfies relative error consistency ($\widehat{\sigma}_n^2 / \sigma_n^2 \xrightarrow{p} 1$), then the test

$$\text{REJECT } H_0 \Leftrightarrow \widehat{R}_n < q_\alpha \widehat{\sigma}_n / \sqrt{n}$$

has asymptotic level α for q_α the α-quantile of a standard normal distribution

Tests for Algorithm Improvement, $k=10$, $\alpha=0.05$

Our CV CLT procedure: valid size, most powerful

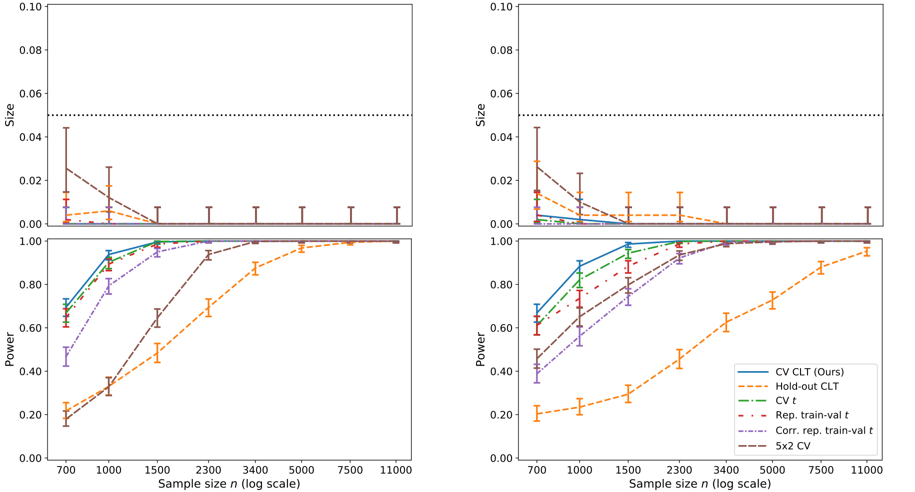


Figure: Size when testing H_1 : $\text{Err}(\mathcal{A}_1) < \text{Err}(\mathcal{A}_2)$ (top) and power when testing H_1 : $\text{Err}(\mathcal{A}_2) < \text{Err}(\mathcal{A}_1)$ (bottom) of level-0.05 tests for improved test error. **Left** (classif.): $\mathcal{A}_1$ = logistic, $\mathcal{A}_2$ = NN. **Right** (reg.): $\mathcal{A}_1$ = RF, $\mathcal{A}_2$ = ridge.

Algorithmic Stability

Stability

How much does the performance of a learned prediction rule change when one point in the training set is changed?

Algorithmic Stability

Stability

How much does the performance of a learned prediction rule change when one point in the training set is changed?

- Uniform stability: worst-case change in loss h_n

Algorithmic Stability

Stability

How much does the performance of a learned prediction rule change when one point in the training set is changed?

- Uniform stability: worst-case change in loss h_n
- Mean-square stability: mean-square change in loss h_n

Algorithmic Stability

Stability

How much does the performance of a learned prediction rule change when one point in the training set is changed?

- Uniform stability: worst-case change in loss h_n
- Mean-square stability: mean-square change in loss h_n
- **Loss stability**
  - Mean-square change in loss *difference* $h_n(Z_0, Z_{1:m}) - \mathbb{E}[h_n(Z_0, Z_{1:m}) \mid Z_{1:m}]$

Algorithmic Stability

Stability

How much does the performance of a learned prediction rule change when one point in the training set is changed?

- Uniform stability: worst-case change in loss h_n
- Mean-square stability: mean-square change in loss h_n
- **Loss stability**
  - Mean-square change in loss *difference* $h_n(Z_0, Z_{1:m}) - \mathbb{E}[h_n(Z_0, Z_{1:m}) \mid Z_{1:m}]$

Note: $\gamma_{loss}(h_n) \leq \gamma_{ms}(h_n)$ [Kumar et al., 2013]

[Bousquet and Elisseeff, 2002, Kale et al., 2011, Kumar et al., 2013, Celisse and Guedj, 2016, . . .]

Asymptotic Normality of CV

CV Central Limit Theorem (Bayle, Bayle, Janson, and Mackey, 2020)

Suppose $Z_0, Z_1, \cdots, Z_n$ are $\mathrm{i.i.d.}$, and define the expected loss function

$$\overline{h}_n(Z_0) = \mathbb{E}[h_n(Z_0, Z_{1:n(1-1/k)}) \mid Z_0] \quad \text{with} \quad \sigma_n^2 = \mathrm{Var}(\overline{h}_n(Z_0)).$$

Asymptotic Normality of CV

CV Central Limit Theorem (Bayle, Bayle, Janson, and Mackey, 2020)

Suppose $Z_0, Z_1, \cdots, Z_n$ are $\mathrm{i.i.d.}$, and define the expected loss function

$$\bar{h}_n(Z_0) = \mathbb{E}[h_n(Z_0, Z_{1:n(1-1/k)}) \mid Z_0] \quad \text{with} \quad \sigma_n^2 = \mathrm{Var}(\bar{h}_n(Z_0)).$$

If loss stability $= o(\sigma_n^2/n)$

Asymptotic Normality of CV

CV Central Limit Theorem (Bayle, Bayle, Janson, and Mackey, 2020)

Suppose $Z_0, Z_1, \cdots, Z_n$ are i.i.d., and define the expected loss function

$$\bar{h}_n(Z_0) = \mathbb{E}[h_n(Z_0, Z_{1:n(1-1/k)}) \mid Z_0] \quad \text{with} \quad \sigma_n^2 = \mathrm{Var}(\bar{h}_n(Z_0)).$$

If loss stability $= o(\sigma_n^2/n)$ and $(\bar{h}_n(Z_0) - \mathbb{E}[\bar{h}_n(Z_0)])^2/\sigma_n^2$ is uniformly integrable

Sufficient condition: $\sup_n \mathbb{E}[|\bar{h}_n(Z_0) - \mathbb{E}[\bar{h}_n(Z_0)]|^\alpha/\sigma_n^\alpha] < \infty$ for some $\alpha > 2$

Asymptotic Normality of CV

CV Central Limit Theorem (Bayle, Bayle, Janson, and Mackey, 2020)

Suppose $Z_0, Z_1, \cdots, Z_n$ are $\mathrm{i.i.d.}$, and define the expected loss function

$$\bar{h}_n(Z_0) = \mathbb{E}[h_n(Z_0, Z_{1:n(1-1/k)}) \mid Z_0] \quad \text{with} \quad \sigma_n^2 = \mathrm{Var}(\bar{h}_n(Z_0)).$$

If loss stability $= o(\sigma_n^2/n)$ and $(\bar{h}_n(Z_0) - \mathbb{E}[\bar{h}_n(Z_0)])^2/\sigma_n^2$ is uniformly integrable then

$$\frac{\sqrt{n}}{\sigma_n}(\widehat{R}_n - R_n) \xrightarrow{d} \mathcal{N}(0,1).$$

Sufficient condition: $\sup_n \mathbb{E}[|\bar{h}_n(Z_0) - \mathbb{E}[\bar{h}_n(Z_0)]|^\alpha/\sigma_n^\alpha] < \infty$ for some $\alpha > 2$

Asymptotic Normality of CV

CV Central Limit Theorem (Bayle, Bayle, Janson, and Mackey, 2020)

Suppose $Z_0, Z_1, \cdots, Z_n$ are $\mathrm{i.i.d.}$, and define the expected loss function

$$\overline{h}_n(Z_0) = \mathbb{E}[h_n(Z_0, Z_{1:n(1-1/k)}) \mid Z_0] \quad \text{with} \quad \sigma_n^2 = \mathrm{Var}(\overline{h}_n(Z_0)).$$

If loss stability $= o(\sigma_n^2/n)$ and $(\overline{h}_n(Z_0) - \mathbb{E}[\overline{h}_n(Z_0)])^2/\sigma_n^2$ is uniformly integrable then

$$\frac{\sqrt{n}}{\sigma_n}(\widehat{R}_n - R_n) \xrightarrow{d} \mathcal{N}(0,1).$$

Sufficient condition: $\sup_n \mathbb{E}[|\overline{h}_n(Z_0) - \mathbb{E}[\overline{h}_n(Z_0)]|^\alpha/\sigma_n^\alpha] < \infty$ for some $\alpha > 2$

Many learning algorithms enjoy decaying loss stability
(e.g., SGD, ERM, k-NN, decision trees, ensemble methods)

Consistent Variance Estimation

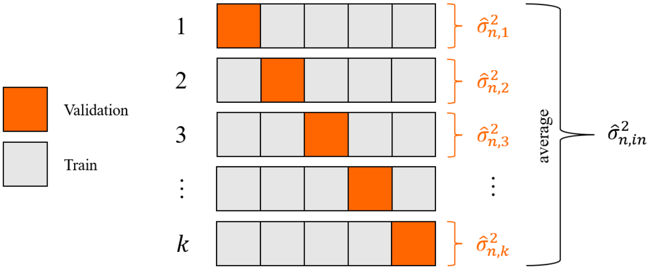
Goal: Find a practical estimator $\widehat{\sigma}_n^2$ satisfying $\widehat{\sigma}_n^2/\sigma_n^2 \xrightarrow{p} 1$ under weak conditions.

Consistent Variance Estimation

Goal: Find a practical estimator $\widehat{\sigma}_n^2$ satisfying $\widehat{\sigma}_n^2/\sigma_n^2 \xrightarrow{p} 1$ under weak conditions.

Within-fold variance estimator $\widehat{\sigma}^2_{n,in}$

Computes the variance of $h_n(Z_i, Z_{B_j})$ in each fold and takes the average across folds



Consistent Variance Estimation

Goal: Find a practical estimator $\widehat{\sigma}_n^2$ satisfying $\widehat{\sigma}_n^2/\sigma_n^2 \xrightarrow{p} 1$ under weak conditions.

All-pairs variance estimator $\widehat{\sigma}^2_{n,out}$

$$\widehat{\sigma}^2_{n,out} = \frac{1}{n}\sum_{j=1}^{k}\sum_{i\in B_j'}(h_n(Z_i, Z_{B_j}) - \widehat{R}_n)^2$$

- Computes the empirical variance of $h_n(Z_i, Z_{B_j})$ across all folds
- Advantage: can also be used for leave-one-out cross-validation

Consistent Variance Estimation

Within-fold variance estimator $\widehat{\sigma}^2_{n,in}$

$$\widehat{\sigma}^2_{n,in} = \frac{1}{k}\sum_{j=1}^{k}\frac{1}{(n/k)-1}\sum_{i\in B'_j}\left(h_n(Z_i, Z_{B_j}) - \frac{k}{n}\sum_{i'\in B'_j} h_n(Z_{i'}, Z_{B_j})\right)^2$$

All-pairs variance estimator $\widehat{\sigma}^2_{n,out}$

$$\begin{aligned}\widehat{\sigma}^2_{n,out} &= \frac{1}{n}\sum_{j=1}^{k}\sum_{i\in B'_j}(h_n(Z_i, Z_{B_j}) - \widehat{R}_n)^2 \\ &= \frac{1}{k}\sum_{j=1}^{k}\frac{k}{n}\sum_{i\in B'_j}(h_n(Z_i, Z_{B_j}) - \widehat{R}_n)^2\end{aligned}$$

Consistent Variance Estimation

Low computational cost

$\widehat{\sigma}^2_{n,in}$ and $\widehat{\sigma}^2_{n,out}$ can be computed in $O(n)$ time, and if loss is binary, in $O(k)$ and $O(1)$ respectively

When h_n is binary, as in the case of 0-1 loss, one can compute

- $\widehat{\sigma}^2_{n,out} = \widehat{R}_n(1 - \widehat{R}_n)$ in $O(1)$ time given access to the overall cross-validation error $\widehat{R}_n$,
- $\widehat{\sigma}^2_{n,in} = \frac{1}{k}\sum_{j=1}^{k} \frac{(n/k)}{(n/k)-1}\widehat{R}_{n,j}(1 - \widehat{R}_{n,j})$ in $O(k)$ time given access to the k average fold errors $\widehat{R}_{n,j} \triangleq \frac{k}{n}\sum_{i\in B'_j} h_n(Z_i, Z_{B_j})$.

Consistent Variance Estimation

Theorem: Consistency of CV Variance Estimators (Bayle et al.)

Under exactly the same conditions given for the CV central limit theorem (loss stability $= o(\sigma_n^2/n)$ and uniform integrability), we have

$$\widehat{\sigma}_{n,in}^2 / \sigma_n^2 \xrightarrow{L^1} 1.$$

Consistent Variance Estimation

Theorem: Consistency of CV Variance Estimators (Bayle et al.)

Under exactly the same conditions given for the CV central limit theorem (loss stability $= o(\sigma_n^2/n)$ and uniform integrability), we have

$$\widehat{\sigma}^2_{n,in}\,/\sigma_n^2 \xrightarrow{L^1} 1.$$

If, additionally, mean-square stability $= o(k\sigma_n^2/n)$, then

$$\widehat{\sigma}^2_{n,out}\,/\sigma_n^2 \xrightarrow{L^1} 1.$$

- Mean-square stability condition particularly mild for leave-one-out CV ($k = n$), becomes $o(\sigma_n^2)$

Conclusions

Summary

- New CV central limit theorem under algorithmic stability

Conclusions

Summary

- New CV central limit theorem under algorithmic stability
- Consistent estimators of CV variance

Conclusions

Summary

- New CV central limit theorem under algorithmic stability
- Consistent estimators of CV variance
- Asymptotically exact confidence intervals and tests for k-fold test error

Conclusions

Summary

- New CV central limit theorem under algorithmic stability
- Consistent estimators of CV variance
- Asymptotically exact confidence intervals and tests for k-fold test error

Thank you!